CS 7545: Machine Learning Theory **Fall 2021**

Lecture 8

Name: Tyler LaBonte **Professor:** Santosh S. Vempala

1 Proof of Spectral Clustering Theorem

Theorem 1.1: Spectral Clustering

Suppose there exists $\mathbf{C}$ with $\|\boldsymbol{C}_i - \boldsymbol{C}_j\| \geq \frac{15k}{\epsilon}\sigma(\mathbf{C})$ and each cluster contains at least ϵn points. Then, the spectral clustering algorithm finds a clustering which differs from $\mathbf{C}$ in at most $\epsilon^2 n$ points. Note that the guarantee is dimension independent.

Proof: Let $\boldsymbol{v}_i$ be the i^{th} row of $\mathbf{A}_k$. Consider the set of "bad" points, which are far from their optimal center:

$$B = \left\{ i : \|\boldsymbol{v}_i - \boldsymbol{c}_i\|_2 \geq \frac{3k}{\epsilon}\sigma(\mathbf{C}) \right\}. \tag{1}$$

Using our lemma, we show that because the Frobenius norm is small, we can't have "too many" bad points.

$$\begin{aligned} 8k\sigma^2(\mathbf{C})n &\geq \|\mathbf{A}_k - \mathbf{C}\|_F^2 && (2)\\ &\geq |B| \left(\frac{3k}{\epsilon}\right)^2 \sigma(\mathbf{C}) && (3)\\ \implies |B| &\leq \frac{8\epsilon^2}{9k} n. && (4) \end{aligned}$$

For points not in B, if i, j are in the same cluster,

$$\|\boldsymbol{v}_i - \boldsymbol{v}_j\| \leq 2 \cdot \frac{3k}{\epsilon}\sigma(\mathbf{C}). \tag{5}$$

If i, j are in different clusters, then by the triangle inequality,

$$\begin{aligned} \|\boldsymbol{v}_i - \boldsymbol{v}_j\| &\geq \|\boldsymbol{c}_i - \boldsymbol{c}_j\| - \|\boldsymbol{v}_i - \boldsymbol{c}_i\| - \|\boldsymbol{v}_j - \boldsymbol{c}_j\| && (6)\\ &\geq \left(\frac{15k}{\epsilon} - \frac{3k}{\epsilon} - \frac{3k}{\epsilon}\right)\sigma(\mathbf{C}) && (7)\\ &= \frac{9k}{\epsilon}\sigma(\mathbf{C}). && (8) \end{aligned}$$

Therefore, when we create the clusters, we should get all "good" points in the cluster and no "good" points outside the cluster, with no guarantee where the "bad" points will end up.

Now we just need that the random point is "good" every iteration. This is slightly subtle since

we remove the clusters each time, but by our assumption that each cluster contains at least ϵn points, the probability we get a "bad" point is at least

$$\left(1 - \frac{\epsilon^2}{k}\right)\left(1 - \frac{\epsilon}{k}\right)^{k-1} \geq 1 - \epsilon, \tag{9}$$

so we have a high-probability guarantee. ■

2 Applications

2.1 Mixture of Gaussians

Suppose F is a mixture of k spherical Gaussians. Then, we need $\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\| \geq \frac{15k}{\epsilon}\sigma$, which has worse dependence on k and ϵ than our earlier algorithm, but is still pretty good.

2.2 Planted Partition

Suppose there is a partition of graph vertices into k components, where $\Pr[e \in C_i] = p$ and $\Pr[e = (u, v) : u \in C_i, v \in C_j] = q$. This is also called the "block model" since the expectation of the adjacency matrix $\mathbb{E}[\mathbf{A}] = \mathbf{C}$ of the graph has a block structure within and between the componenets. We'd like to recover the partition with high probability.

We can interpret the rows of $\mathbf{C}$ as the "centers" of the components (note $\mathbf{C}$ is rank k). For i, j in different components,

$$\|\boldsymbol{c}_i - \boldsymbol{c}_j\|^2 = (p - q)^2 \cdot \frac{2n}{k}. \tag{10}$$

To find $\sigma(\mathbf{C})$ we need $\|\mathbf{A} - \mathbb{E}[\mathbf{A}]\|_2$. We will use the following theorem from random matrix theory:

Theorem 2.1: Variance of a Random Matrix

Suppose $\mathbf{R}$ is a random matrix with independent entries, $\mathbb{E}[R_{ij}] = 0$, and $\mathrm{Var}(R_{ij}) = \sigma^2$. Then with high probability, $\|\mathbf{R}\|_2 \leq (2 + o(1))\sigma\sqrt{n}$.

Thus, since $\sigma^2 \leq p$, we have $\|\mathbf{A} - \mathbb{E}[\mathbf{A}]\|_2 \leq 3\sqrt{pn}$. So,

$$\sigma^2(\mathbf{C}) = \frac{1}{n}(3\sqrt{pn})^2 = 9p. \tag{11}$$

Applying the spectral clustering theorem, it suffices to have

$$(p - q)^2 \cdot \frac{2n}{k} \geq 9p\left(15\frac{k}{\epsilon}\right)^2 \tag{12}$$

to correctly cluster $(1 - \epsilon^2)n$ points. If we set $p = a/n$ and $q = b/n$ as is common, we obtain

$$a - b \geq \frac{ck^{3/2}}{\epsilon}\sqrt{a}, \tag{13}$$

which essentially only scales as $\sqrt{a}$ for fixed k.

The information-theoretic limit for this problem is $(a-b)^2 > 2a$, so the spectral clustering algorithm is pretty good!

2.3 Planted Clique

Here is a similar problem. Suppose $G = G_{n,\frac{1}{2}}$ is a random graph on n nodes that includes each edge independently with probability $\frac{1}{2}$. We'd like to find the largest clique in G. This is a very hard problem for general graphs, and is in fact **NP**-hard to approximate within $n^{1-\epsilon}$ for any $\epsilon > 0$. However, for a random graph, we have the following theorem.

Theorem 2.2: Planted Clique in Random Graphs

With high probability, $G_{n,\frac{1}{2}}$ has a max clique of size $(2 + o(1)) \log n$.

We can find a $\log n$-sized clique by recursively picking highest degree nodes on a connected subgraph.

Consider a planted clique with $k \gg 2\log(n)$. Clique nodes have expected degree $\frac{1}{2}(n+k)$ as opposed to the typical $\frac{n}{2}$. With high probability, a typical node has degree $\frac{n}{2} \pm \sqrt{2n \log n}$ by Chernoff. So, if $k > \sqrt{n \log n}$, we can find a the clique just by taking the highest degree nodes.

However, we can find smaller clique by using the spectral algorithm.

Theorem 2.3: Spectral Planted Clique

Using the spectral clustering algorithm where we let $\boldsymbol{v}$ be the top singular vector of $\mathbf{A}$ and take all vertices connected to the top $3k/4$ entries of $\boldsymbol{v}$, we can find any clique of size $k \geq c\sqrt{n}$.